CLAIMS:

What is claimed is:

1. A multithreading processor, comprising:

a thread control unit;

a multithreaded register file having a plurality of registers; and

a plurality of hold latches, wherein

each of a plurality of the registers in the multithreaded register file and each of a plurality of the hold latches stores data representing a first instruction thread and a second instruction thread; and

the thread control unit provides a thread control signal to said hold latches and registers selecting a thread using said data.

2. The multithreading processor as recited in claim 1, wherein the thread control unit via the thread control signal places at least one of the plurality of the hold latches and at least one of the plurality of the register files into an interleaving multithreading mode.

3. The multithreading processor as recited in claim 1, wherein the thread control unit, responsive to a determination that a latency in an instruction exceeding a first predetermined time has occurred in one of the two threads, sends control signals to said hold latches and register files for reading out data exclusively from the other of the two threads until a second predetermined time has elapsed.

4. The multithreading processor as recited in claim 3, wherein the thread control unit returns the plurality of hold latches and register files to an interleaving multithreading mode after the expiration of the second time period.

5. The multithreading processor as recited in claim 3, wherein the latency in an instruction exceeding a first predetermined time results from a load instruction that misses in a datacache.

6. The multithreading processor as recited in claim 3, wherein the latency in an instruction exceeding a first predetermined time results from a mispredicted branch.

7. A data processing system, comprising:

a memory unit;

a mixed-mode multithreading processor; and

a bus coupling the multimedia multithreading processor; wherein

the multimedia multithreading processor comprises:

a thread control unit;

a multithreaded register file having a plurality of registers; and

a plurality of hold latches; wherein

each of a plurality of the registers in the multithreaded register file and each of a plurality of the hold latches stores data representing a first instruction thread and a second instruction thread; and

the thread control unit provides thread control signals to said hold latches and registers selecting a thread using said data.

8. The data processing system as recited in claim 7, wherein the thread control unit via the thread control signal places at least one of the plurality of the hold latches and at least one of the plurality of the registers into an interleaving multithreading mode.

9. The data processing system as recited in claim 7, wherein the thread control unit, responsive to a determination that a latency in an instruction exceeding a first predetermined time has occurred in one of the two threads, sends control signals to said hold latches and registers for reading out data exclusively from the other of the two threads until a second predetermined time has elapsed.

10. The data processing system as recited in claim 9, wherein the thread control unit returns said hold latches and registers to an interleaving multithreading mode after the expiration of the second time period.

11. The data processing system as recited in claim 9, wherein the latency in an instruction exceeding a first predetermined time results from a load instruction that misses in a datacache.

12. The data processing system as recited in claim 9, wherein the latency in an instruction exceeding a first predetermined time results from a mispredicted branch.

13. The data processing system as recited in claim 7, wherein the multimedia multithreading processor is a first multimedia multithreading processor, the thread control unit is a first thread control unit, the hold latches are first hold latches, the multithreaded register file is a first multithreaded register file, the plurality of registers are a plurality of first registers, and the data is a first data, and further comprising:

a second multimedia multithreading processor, wherein the second multimedia multithreading processor comprises:

a second thread control unit;

a second multithreaded register file having a plurality of second registers; and

a plurality of second hold latches, wherein each of a plurality of the second registers and each of a plurality of the second hold latches stores second data representing a third instruction thread and a fourth instruction thread; and

the second thread control unit provides second thread control signals to said second hold latches and second registers selecting a thread using said second data.

14. A processor for use in a data processing system, the processor comprising:

a plurality of flow through latches; and

a plurality of hold state latches, wherein

the hold state latches store two data units, with one data unit corresponding to a first thread and a second data unit corresponding to a second thread, and

control signals determine which of the two data units is read out of each of the plurality hold state latches.

15. The processor as recited in claim 14, wherein, responsive to a determination that one thread is not active, reading data corresponding to only one of the threads for a period of time.

16. The processor as recited in claim 15, wherein the period of time is a predetermined amount of time corresponding to a predicted latency in the one thread that is not active.

17. A data processing system, comprising:

a memory unit;

a mixed-mode multithreading processor; and

a bus coupling the multimedia multithreading processor; wherein

the multimedia multithreading processor comprises:

a plurality of flow through latches; and

a plurality of hold state latches, wherein

the hold state latches store two data units, with one data unit corresponding to a first thread and a second data unit corresponding to a second thread, and

control signals determine which of the two data units is read out of each of the plurality hold state latches.

18. The data processing system as recited in claim 17, wherein, responsive to a determination that one thread is not active, reading data corresponding to only one of the threads for a period of time.

19. The data processing system as recited in claim 18, wherein the period of time is a predetermined amount of time corresponding to a predicted latency in the one thread that is not active.